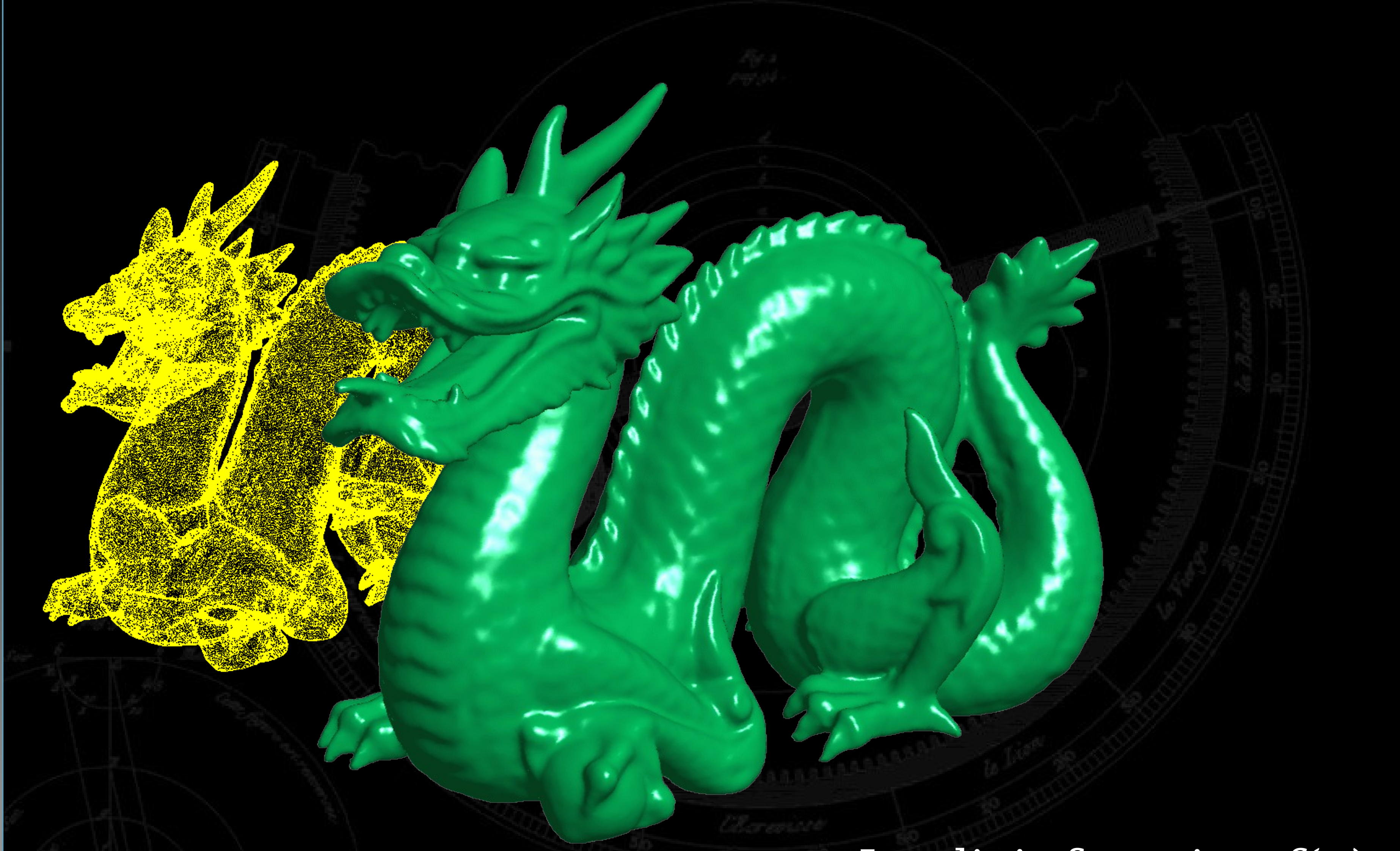
Reconstruction and Representation of 3D Objects with Radial Basis Functions

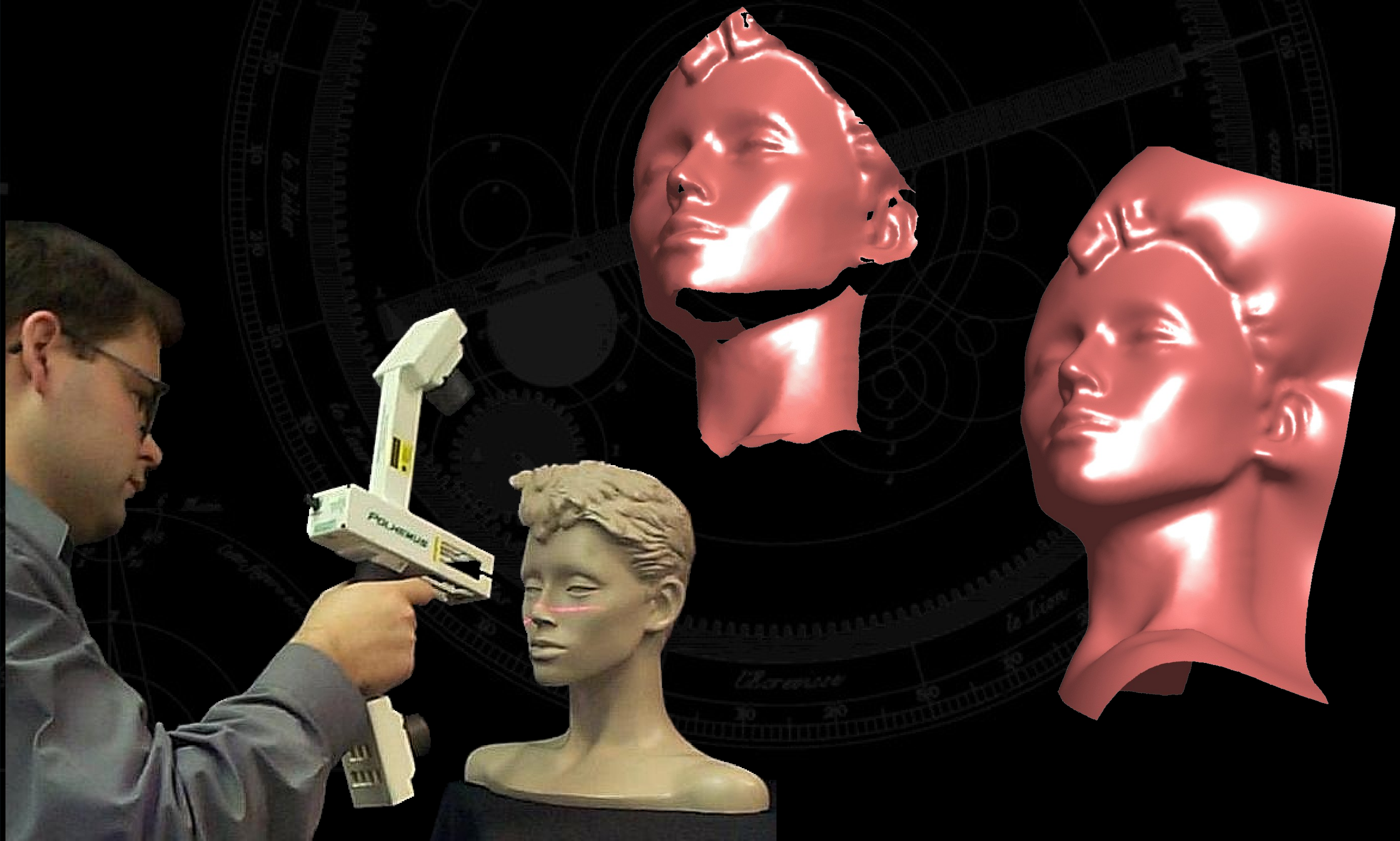
J. C. Carr[1,2], R. K. Beatson[2], J. B. Cherrie[1,2], T. J. Mitchell[1,2], W. R. Fright[1], B. C. McCallum[1] & T. R. Evans[1]

[1]Applied Research Associates NZ Ltd,
[2]Dept. Mathematics & Statistics, University of Canterbury, NZ.



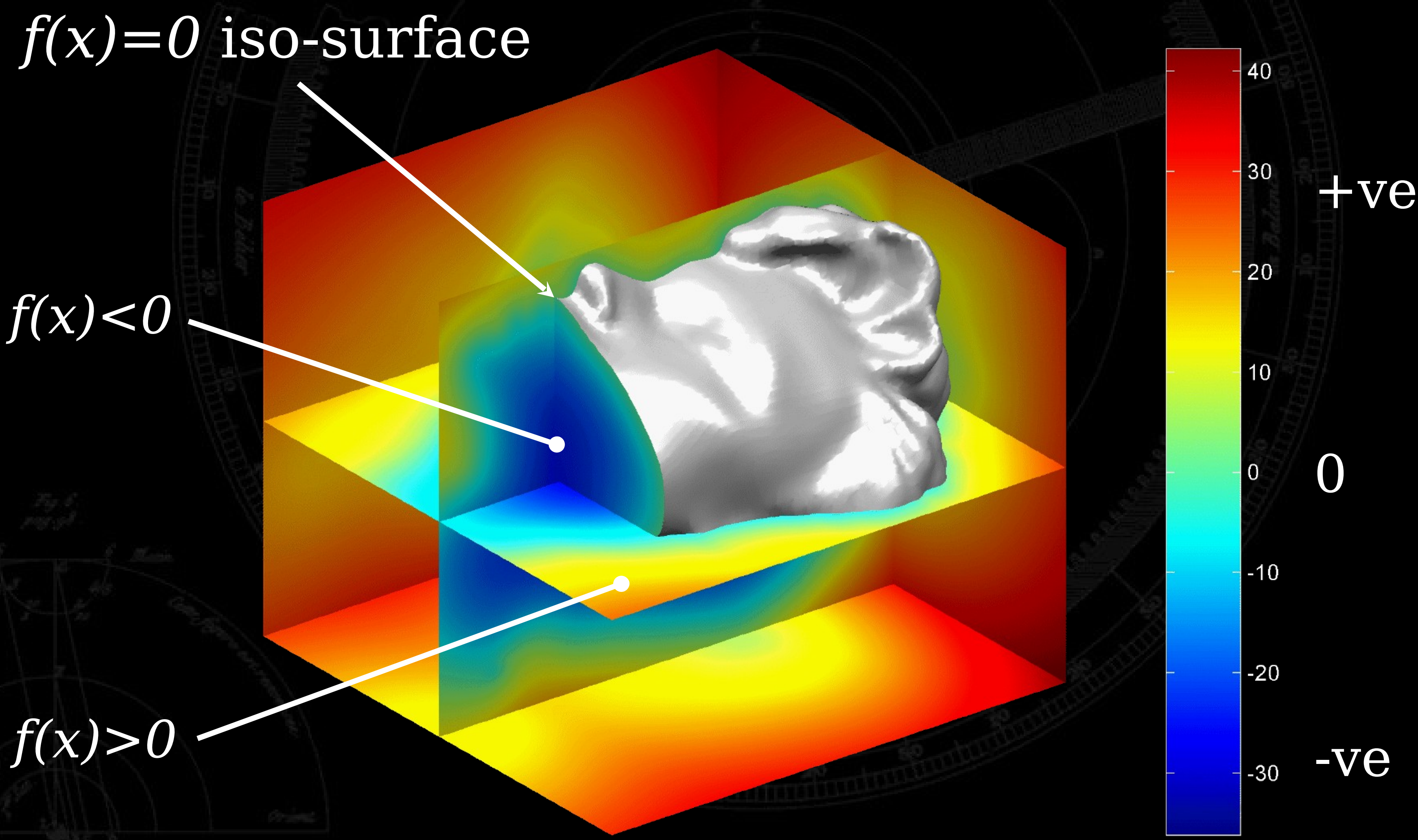
472,000 point-cloud → Implicit function *f(x)*
(32,000 term RBF)

Laser scanning & mesh repair



Mesh repair

Implicit surface modeling



RBF surface modeling

The problem

To find an interpolant *s* such that

$s(x_i) = 0, \qquad i = 1,\ldots,n$ (known surface poi

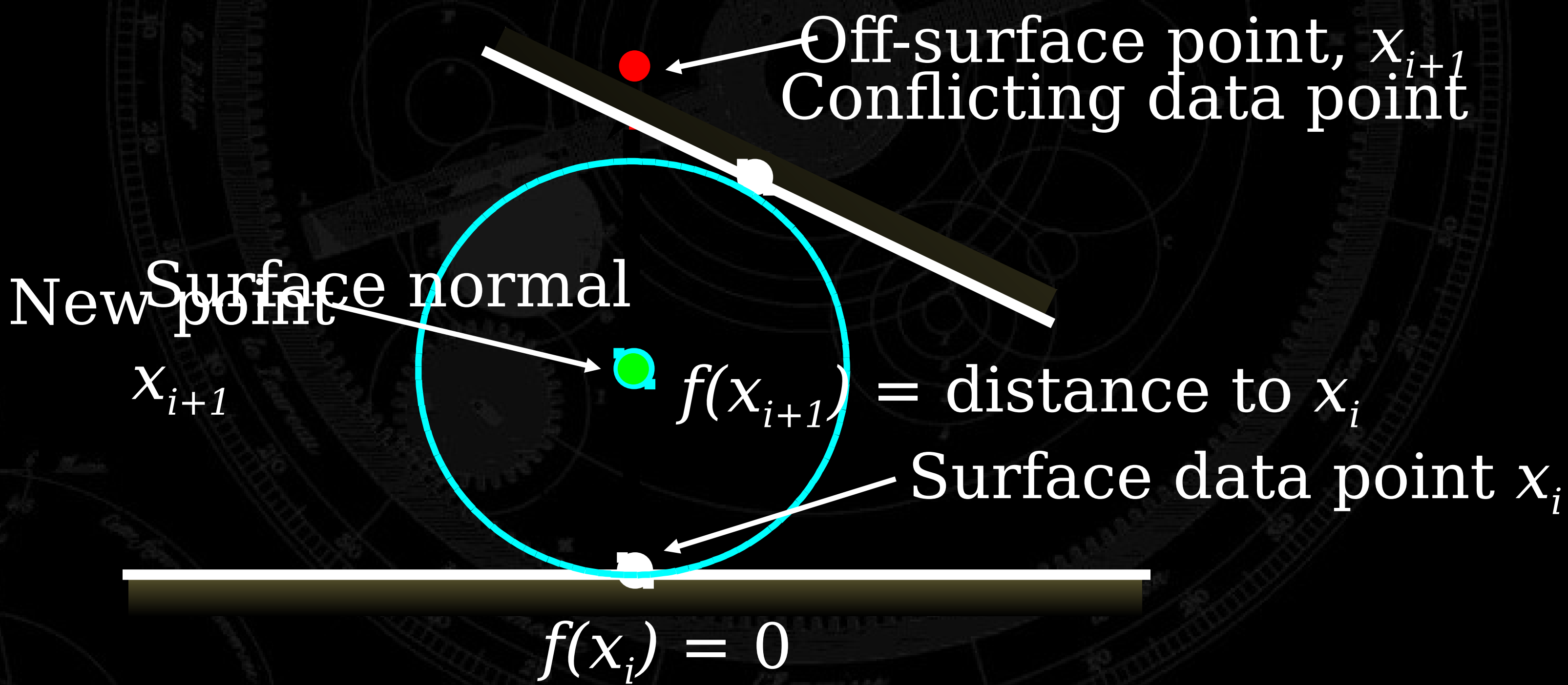
$s(x_i) = d_i \neq 0, \qquad i = n+1,\ldots,N$ (*off*-surface point

Our method

- Form a signed-distance distribution
- Interpolate distance field (fit an RBF)
- Iso-surface RBF

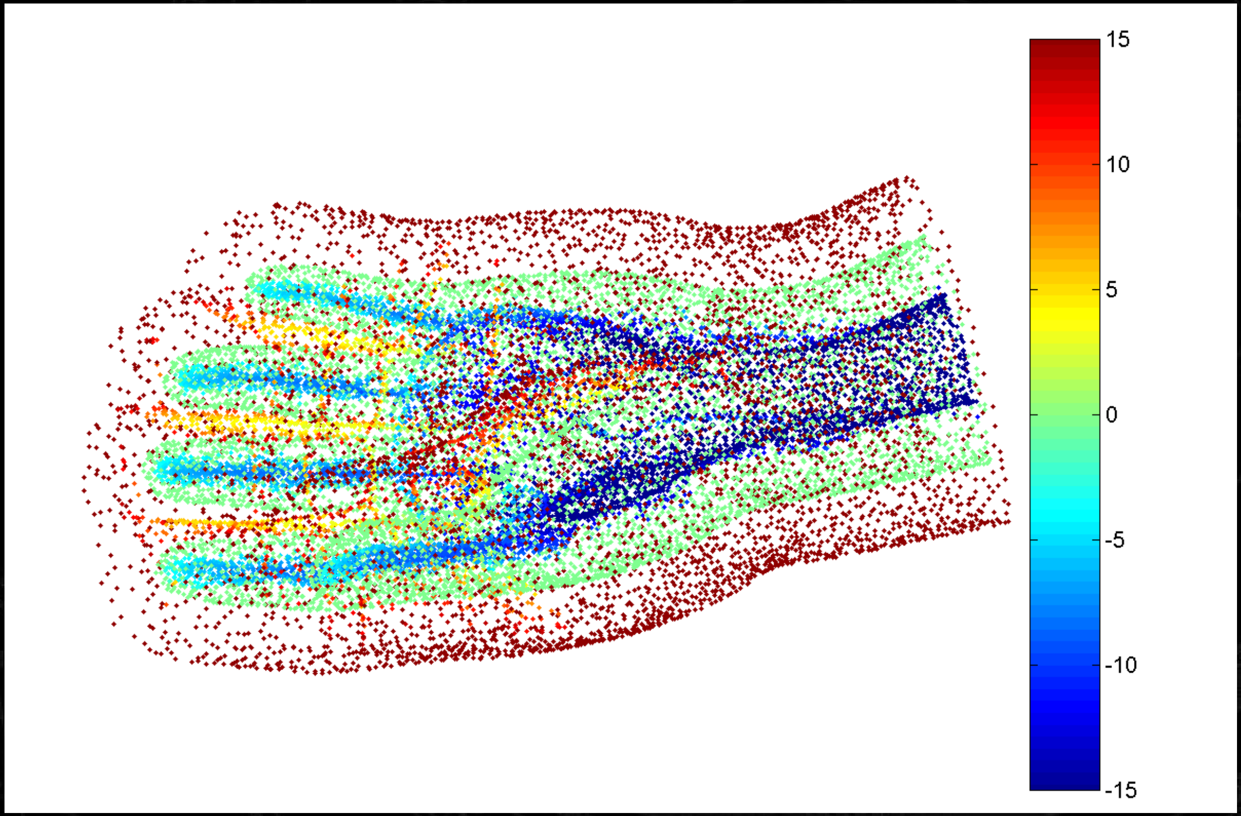
Generating off-surface data

Ensure a consistent distance-to-surface field



- Validate normal lengths

Forming a signed-distance function



Outward normal points

On-surface points

Inward normal points

- Off-surface points are projected along surface normals

Minimal energy interpolants

We want to find the *smoothest* function which fits our distance-surface data.

$$\text{minimize} \int_{\mathbb{R}^3} \left(\frac{\partial^2 s(\boldsymbol{x})}{\partial x^2}\right)^2 + \left(\frac{\partial^2 s(\boldsymbol{x})}{\partial y^2}\right)^2 + \left(\frac{\partial^2 s(\boldsymbol{x})}{\partial z^2}\right)^2$$
$$+ 2\left(\frac{\partial^2 s(\boldsymbol{x})}{\partial x \partial y}\right)^2 + 2\left(\frac{\partial^2 s(\boldsymbol{x})}{\partial x \partial z}\right)^2 + 2\left(\frac{\partial^2 s(\boldsymbol{x})}{\partial y \partial z}\right)^2 d\boldsymbol{x}.$$

i.e., minimize the 2[nd] derivative

Thin-plate spline in 3D

The minimizing interpolant has the form :

Linear polynomial

Scalar weight

$$s(\boldsymbol{x}) = p(\boldsymbol{x}) + \sum_{i=1}^{N} \lambda_i |\boldsymbol{x} - \boldsymbol{x}_i|,$$

How far is ***x*** from $\boldsymbol{x}_i$

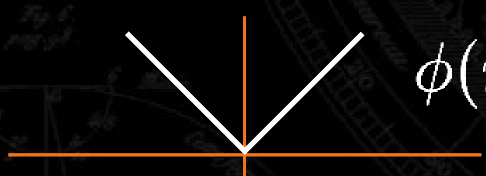
Radial Basis Functions

This is a specific example of an RBF

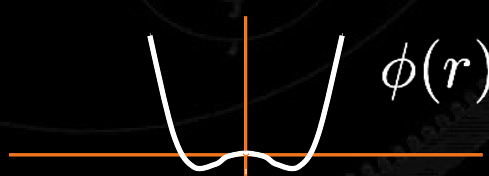
$$s(\boldsymbol{x}) = p(\boldsymbol{x}) + \sum_{i=1}^{N} \lambda_i \phi(|\boldsymbol{x} - \boldsymbol{x}_i|),$$

Choices for φ

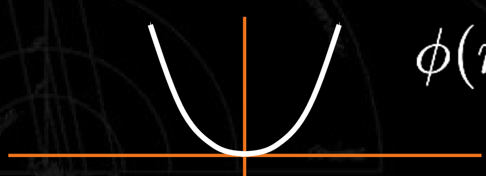
where $\phi(r) = |r|$ Minimizes 2nd derivative in 3D



$\phi(r) = |r|$



$\phi(r) = r^2 \log(r)$



$\phi(r) = r^3$ Minimizes 3nd derivative in 3D

Minimizes 2nd derivative in 2D

How do we find the weights λ_i ?

Form & solve the linear system :

N+4 x N+4

$$\begin{pmatrix} A & P \\ P^{\mathsf{T}} & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{\lambda} \\ \boldsymbol{c} \end{pmatrix} = \begin{pmatrix} \boldsymbol{f} \\ 0 \end{pmatrix}$$

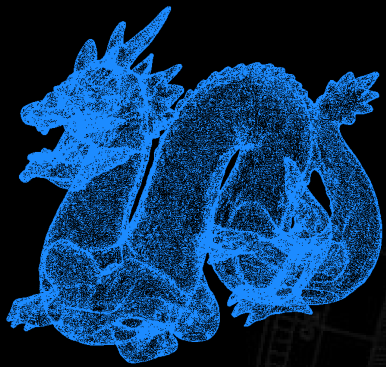
Matrix dependent on the locations of the data points

Unknowns to find

Known distance values at x_i

where $A_{i,j} = \phi(|\boldsymbol{x}_i - \boldsymbol{x}_j|), \qquad i,j = 1,\ldots,N,$

$$P_{i,j} = p_j(\boldsymbol{x}_i), \qquad i = 1,\ldots,N, \quad j = 1,\ldots,\ell.$$



Fast solution

N+4 x N+4

$$\begin{pmatrix} A & P \\ P^{\mathsf{T}} & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{\lambda} \\ \boldsymbol{c} \end{pmatrix} = \begin{pmatrix} \boldsymbol{f} \\ 0 \end{pmatrix}$$

	Direct methods	**Fast methods**
storage	$N\,(N+1)$	$O(N)$
flops	$N^3/6 + O(N^2)$	$O(N \log N)$

E.g. dragon : 3,000 GB (N=872,000) 0.3 GB

E.g. dragon : not possible 2:51:00 (PIII 550MHz)

Fast evaluation

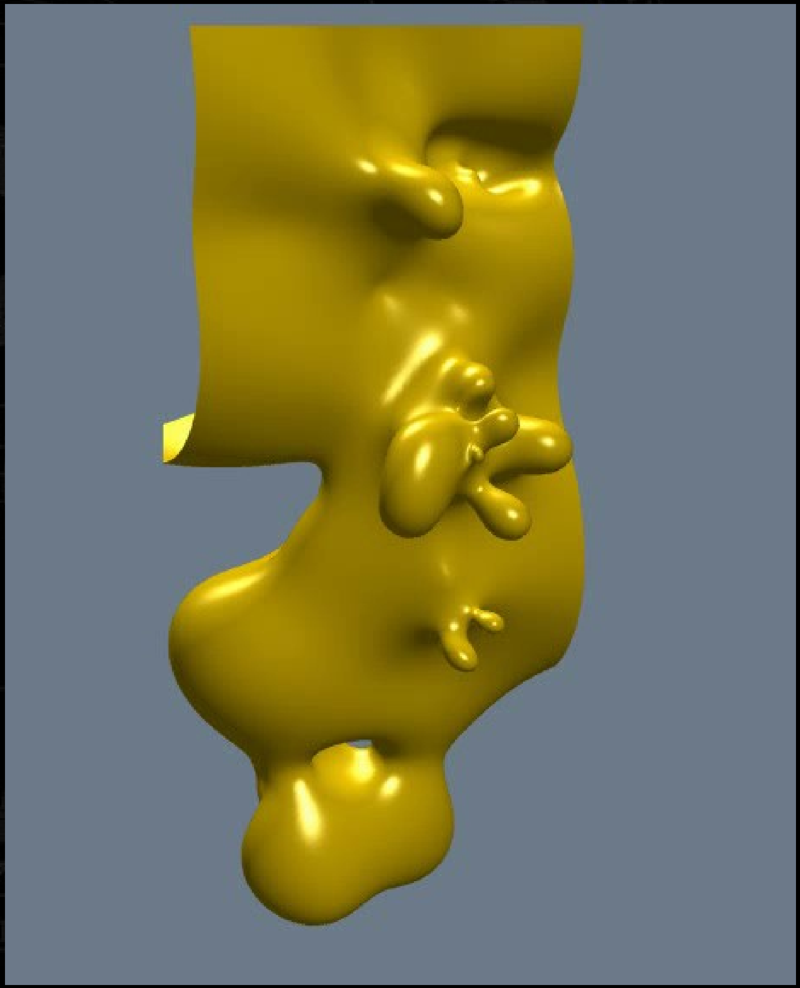
$$s(\boldsymbol{x}) = p(\boldsymbol{x}) + \sum_{i=1}^{N} \lambda_i \phi(|\boldsymbol{x} - \boldsymbol{x}_i|)$$

	Direct methods	**Fast methods**
flops per evaluation	*O(N)*	*O(1) + O(N log N) setup*

Centre reduction

Greedy algorithm

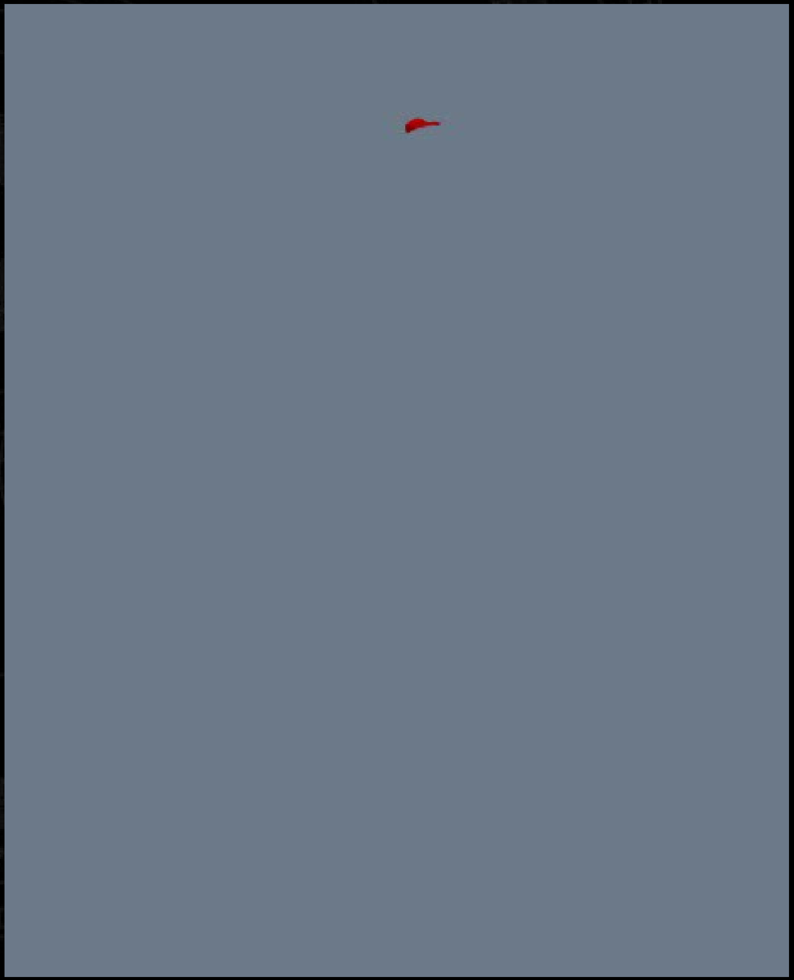
- Fit an RBF to a subset of the x_i
- Evaluate $\varepsilon_i = f_i - s(x_i)$ at *all* the nodes
- If $\max|\varepsilon_i| < \varepsilon_{fit\_acc}$ stop
- else add centres where ε_i is large
- re-fit RBF



1,086,000 points ⇒ 82,000 centres

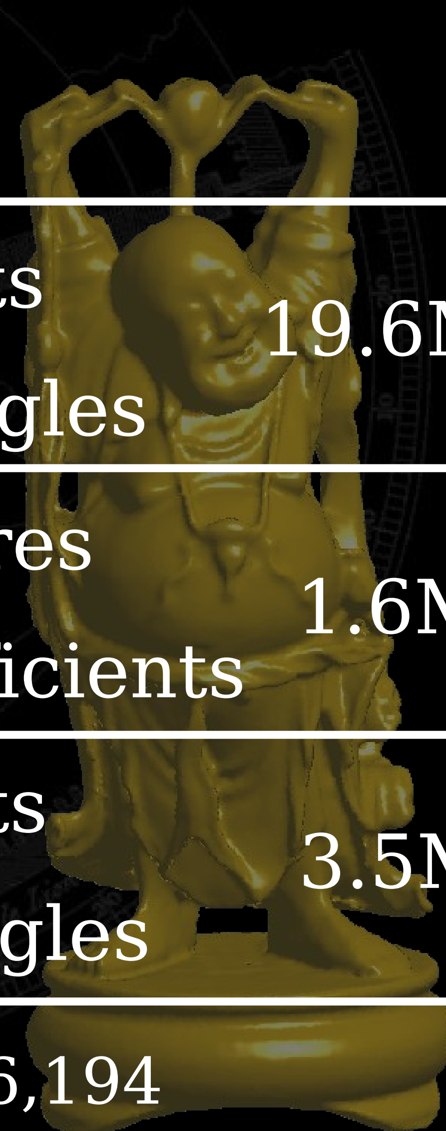
Iso-surfacing

- Surface-following minimizes RBF evaluations
- RBF centres are used as seeds
- The RBF gradient assists seeding and mesh optimisation



Results

Buddha

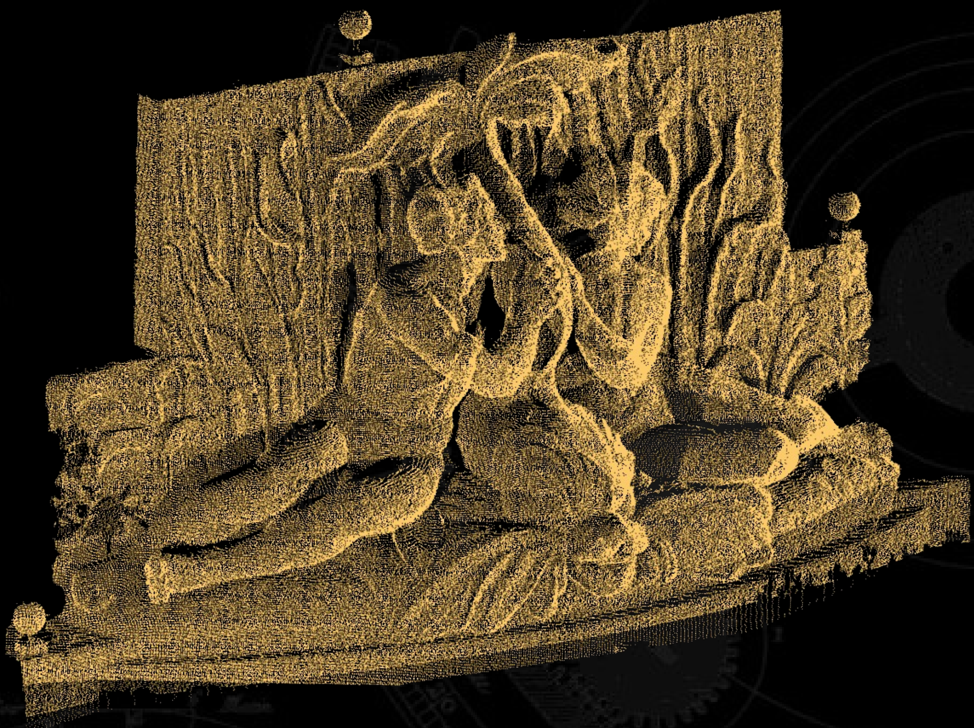


Original mesh	543 652	points	19.6MB
	1 086 798	triangles	
RBF representation	80 518	centres	1.6MB
	80 522	coefficients	
New mesh	96 766	points	3.5MB
	193 604	triangles	

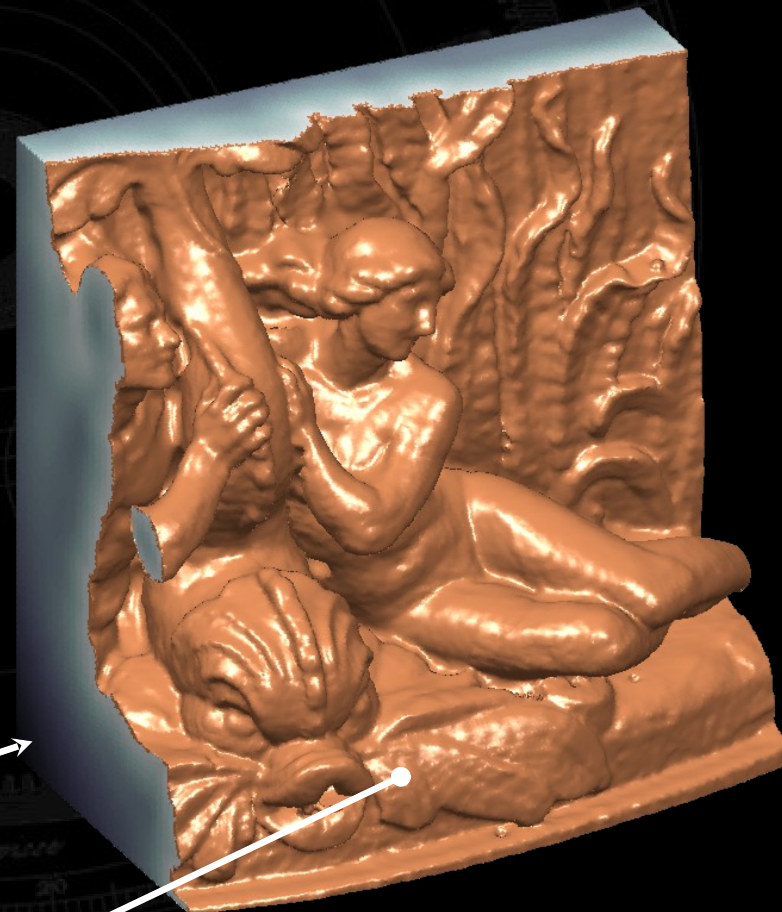
\# interpolation points: 1,086,194

Fit time: 4:03:26 Eval time: 0:04:07 (500MHz PIII)

Interpolating noisy data



350,000 point LIDAR scan



RBF distance field

Zero-valued iso-surface

Spline smoothing

Look for the function $s^\star$ that minimizes

$$\rho\|s\|^2 + \frac{1}{N}\sum_{i=1}^{N}\left(s(\boldsymbol{x}_i) - f_i\right)^2$$

2nd derivative energy term (smoothness)

Closeness of fit

$s^\star$ is also an RBF with the coefficients given by

$$\begin{pmatrix} A - 8N\pi\rho I & P \\ P^{\mathsf{T}} & 0 \end{pmatrix}\begin{pmatrix} \boldsymbol{\lambda} \\ \boldsymbol{c} \end{pmatrix} = \begin{pmatrix} \boldsymbol{f} \\ 0 \end{pmatrix}$$

$$\|s\|^2 = \int_{\mathbb{R}^3} \left(\frac{\partial^2 s(\boldsymbol{x})}{\partial x^2}\right)^2 + \left(\frac{\partial^2 s(\boldsymbol{x})}{\partial y^2}\right)^2 + \left(\frac{\partial^2 s(\boldsymbol{x})}{\partial z^2}\right)^2 + 2\left(\frac{\partial^2 s(\boldsymbol{x})}{\partial x \partial y}\right)^2 + 2\left(\frac{\partial^2 s(\boldsymbol{x})}{\partial x \partial z}\right)^2 + 2\left(\frac{\partial^2 s(\boldsymbol{x})}{\partial y \partial z}\right)^2 d\boldsymbol{x}.$$

Spline smoothing

Look for the function $s^\star$ that minimizes

$$\rho \|s\|^2 + \frac{1}{N}\sum_{i=1}^{N}\left(s(\boldsymbol{x}_i) - f_i\right)^2$$

Rearranging :

Deviation at each data point

$$\begin{pmatrix} A & P \\ P^\top & 0 \end{pmatrix}\begin{pmatrix} \boldsymbol{\lambda} \\ \boldsymbol{c} \end{pmatrix} = \begin{pmatrix} \boldsymbol{f} - 8N\pi\rho\boldsymbol{\lambda} \\ 0 \end{pmatrix}$$

ρ determines amount of smoothing

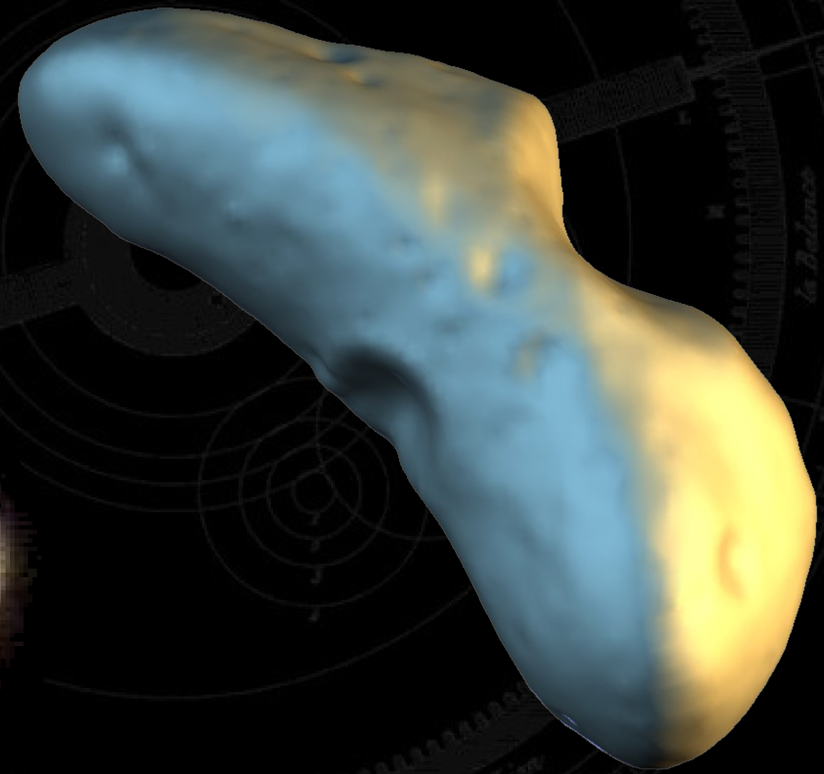
Spline smoothing with RBFs

Exact fit
($\rho = 0$)

Increasing ρ

Increasing smoothness

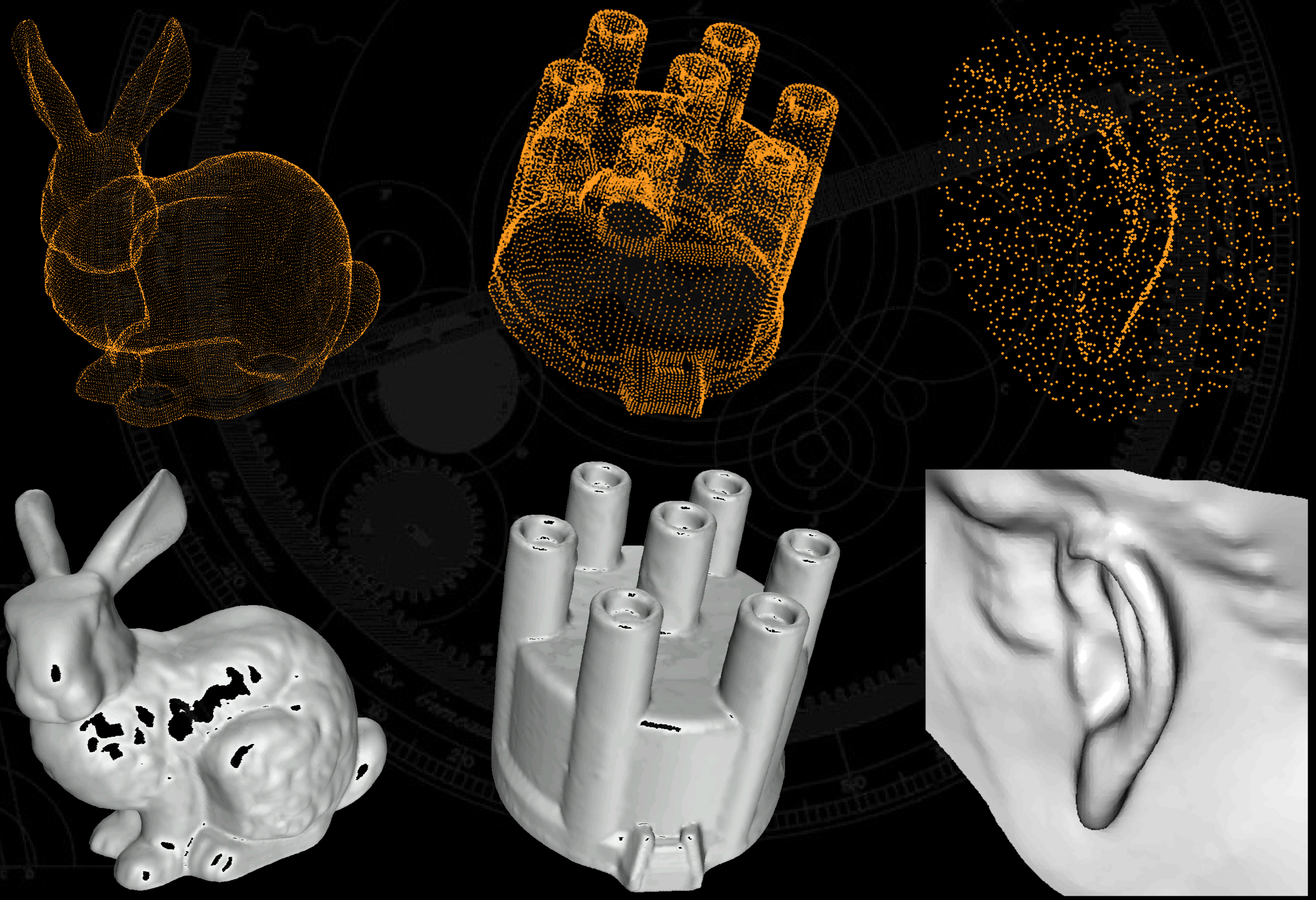
Reconstructing Eros



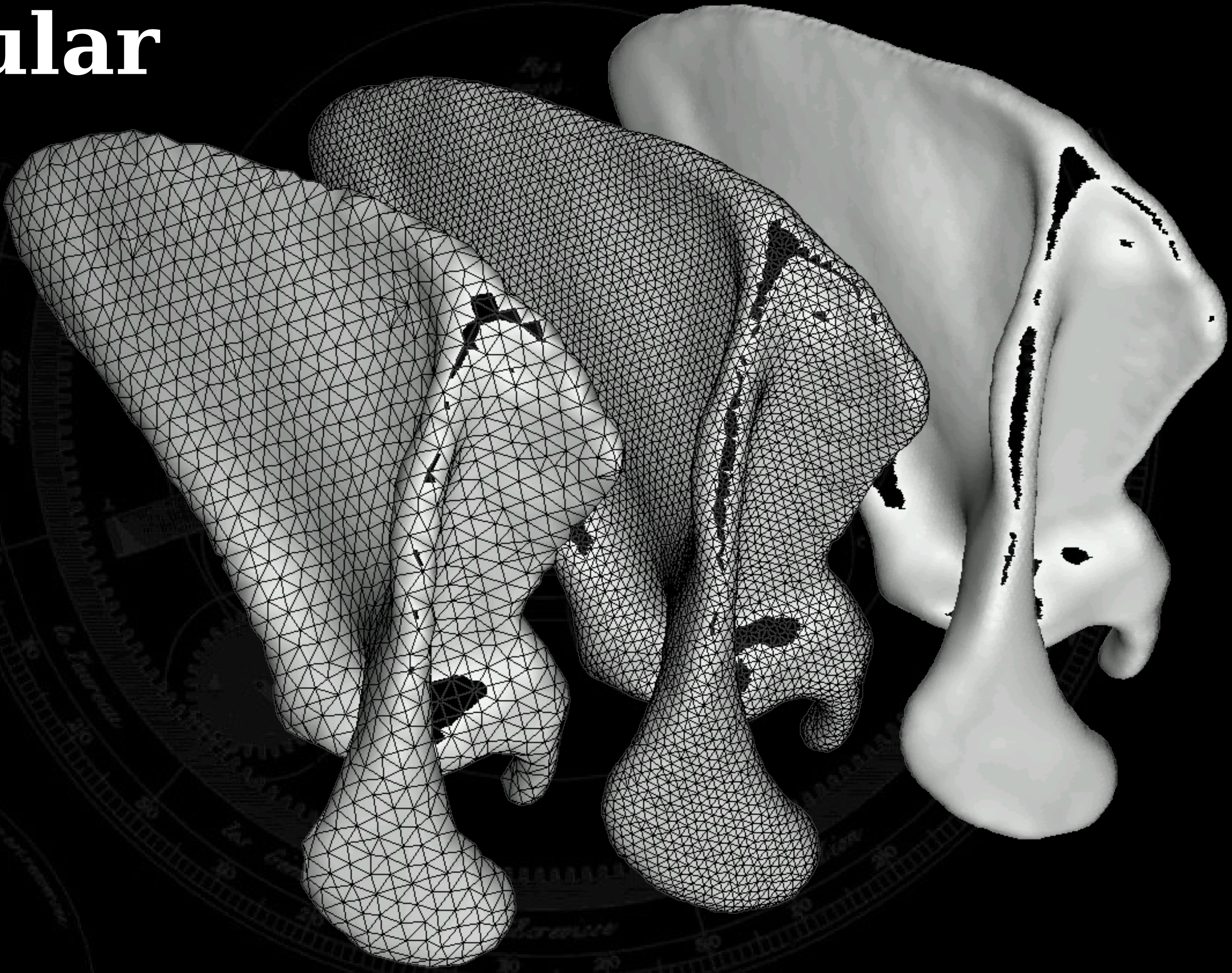
Interpolating irregular, noisy, non-uniformly sampled range data from NASA's NEAR spacecraft

http://near.jhuapl.edu/iod/20000728/index.html

Surfacing examples

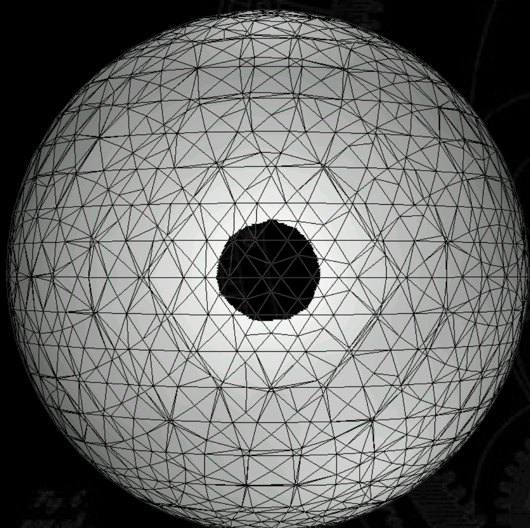


Scapular

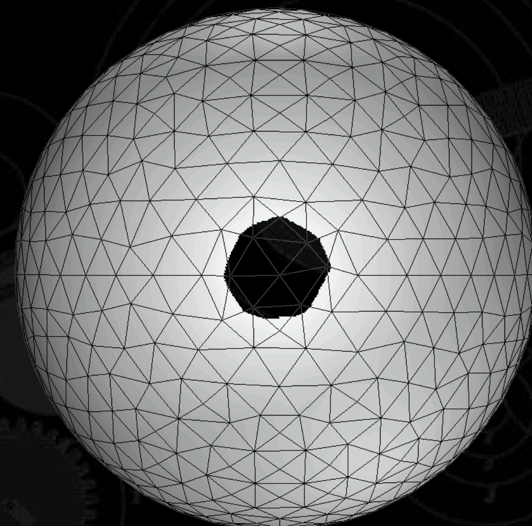


- Evaluate mesh at the desired resolution

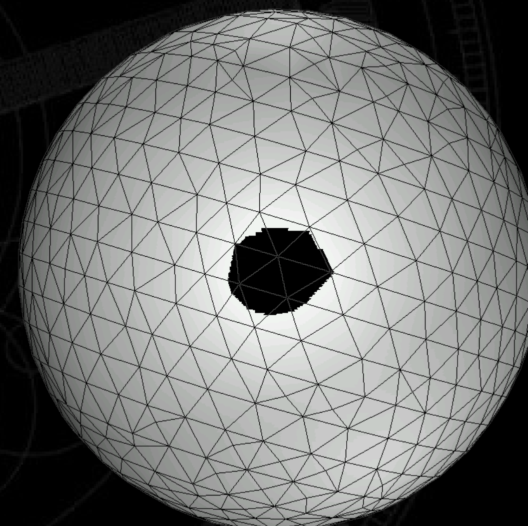
Mesh optimisation



Grid-constrained mesh



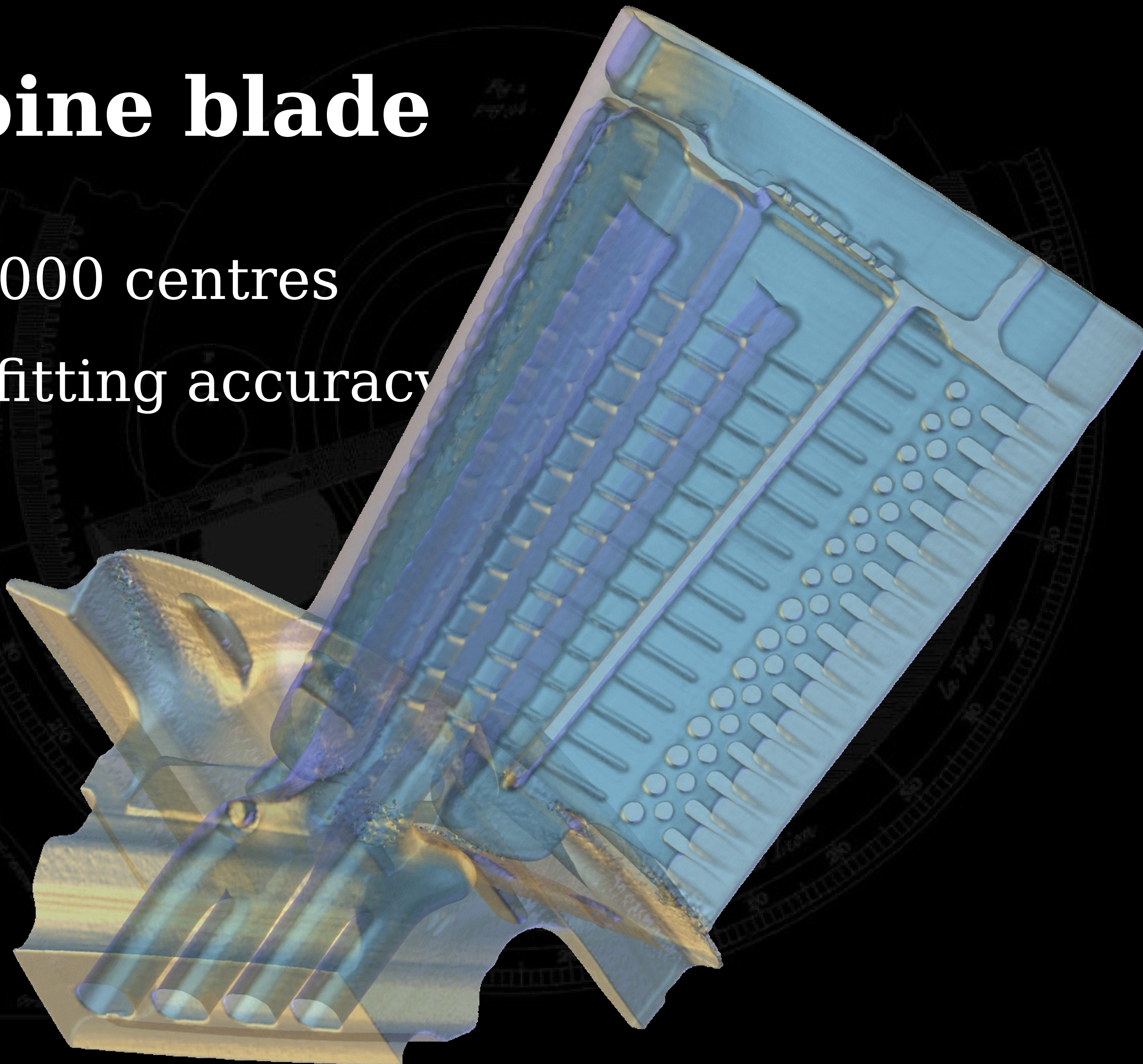
Optimised mesh



Mesh vertices constrained to lie on parallel planes

Turbine blade

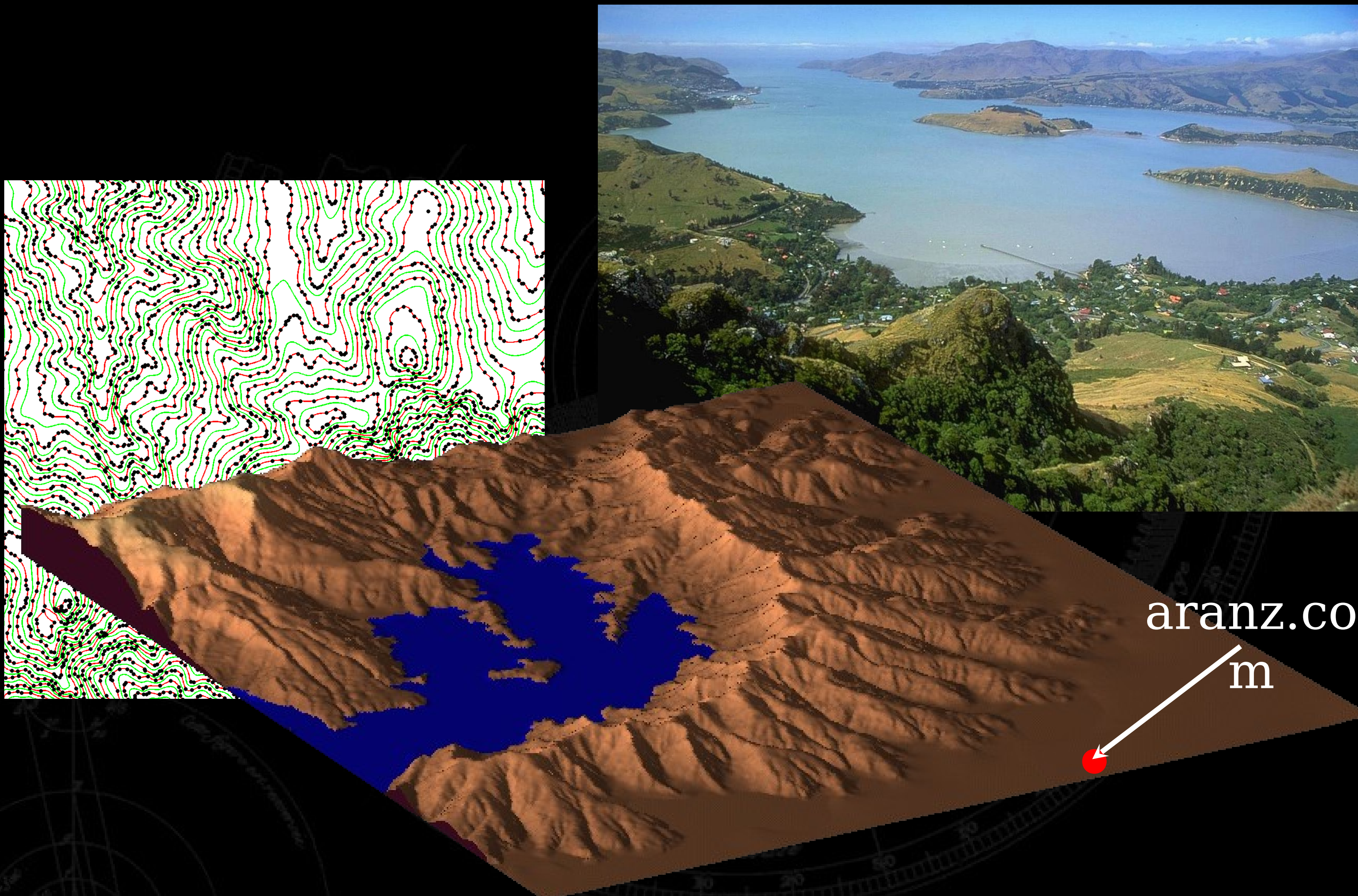
- 594,000 centres
- 10^{-4} fitting accuracy



Conclusions

- A functional representation of a complex object is possible i.e. $f(x)$
- Smooth RBF interpolation is ideal for mesh repair
- The smoothest surface, most consistent with the input data, is produced
- Gradients are determined analytically, i.e. $\nabla f(x)$
- Fast evaluation is essential

Email : j.carr@aranz.com



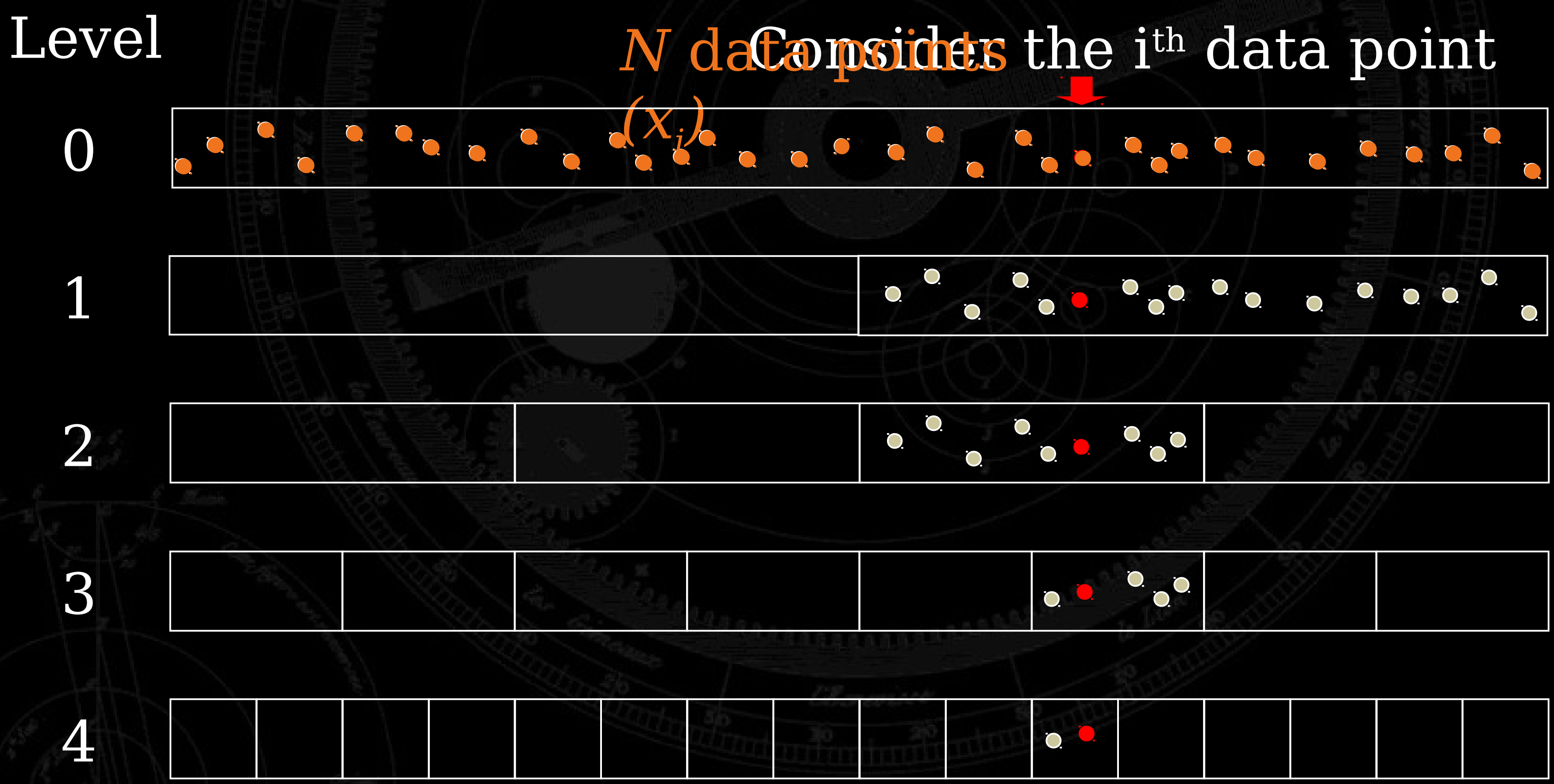
RBF model of Christchurch & Lyttelton $\phi(r) = r^2 \log(r)$

Fast multipole methods

Fast multipole methods

1-D fast evaluation example

1. Hierarchical partitioning of space



Fast multipole methods

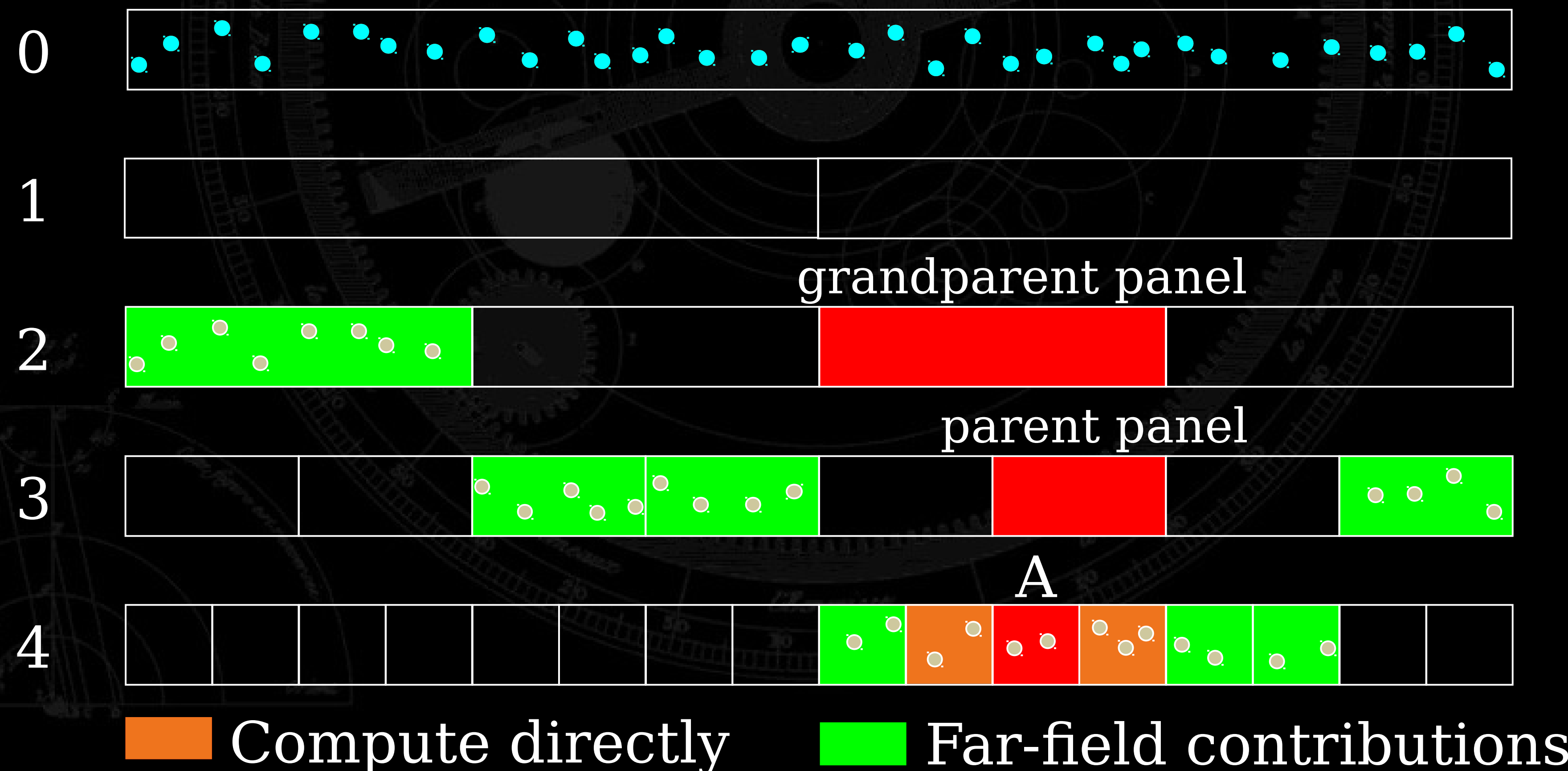
2. Approximate far-panel contributions with Laurent se

Compute near panel contributions directly

Consider evaluating *s(x)* inside panel A

Evaluation cost : *O(log N) + setup*

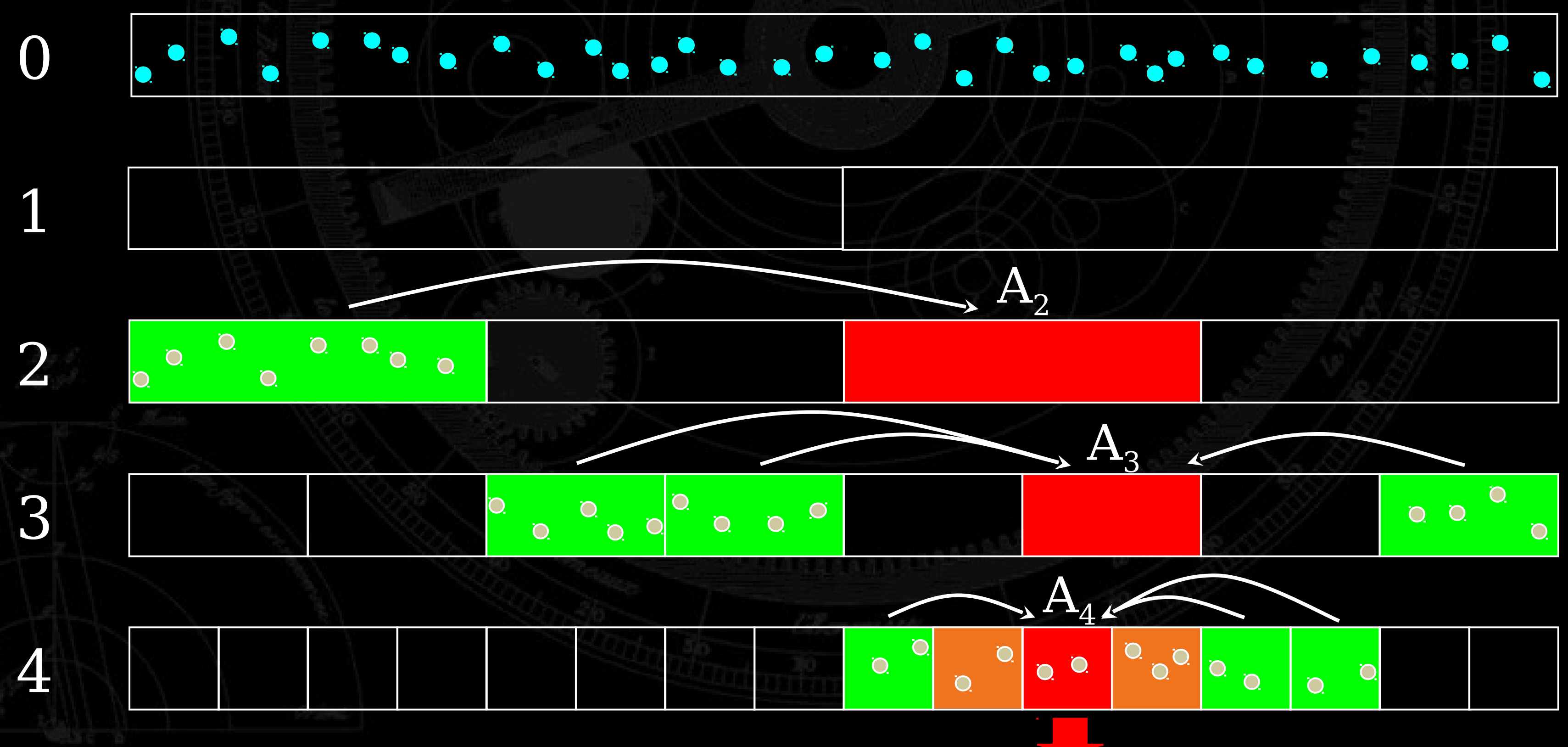
Level



Fast multipole methods

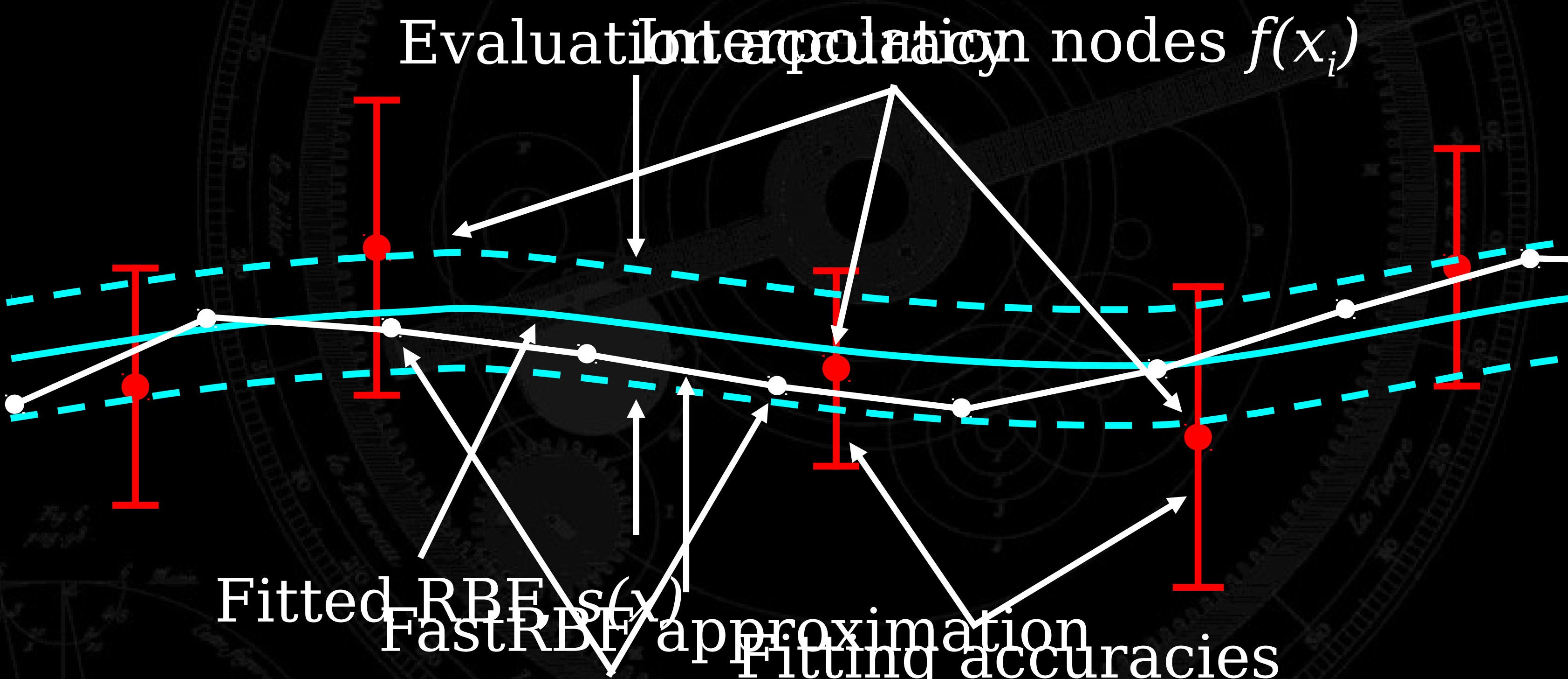
Level

Evaluation cost : *O(1)* + *setup*



P(x) Taylor series approximation

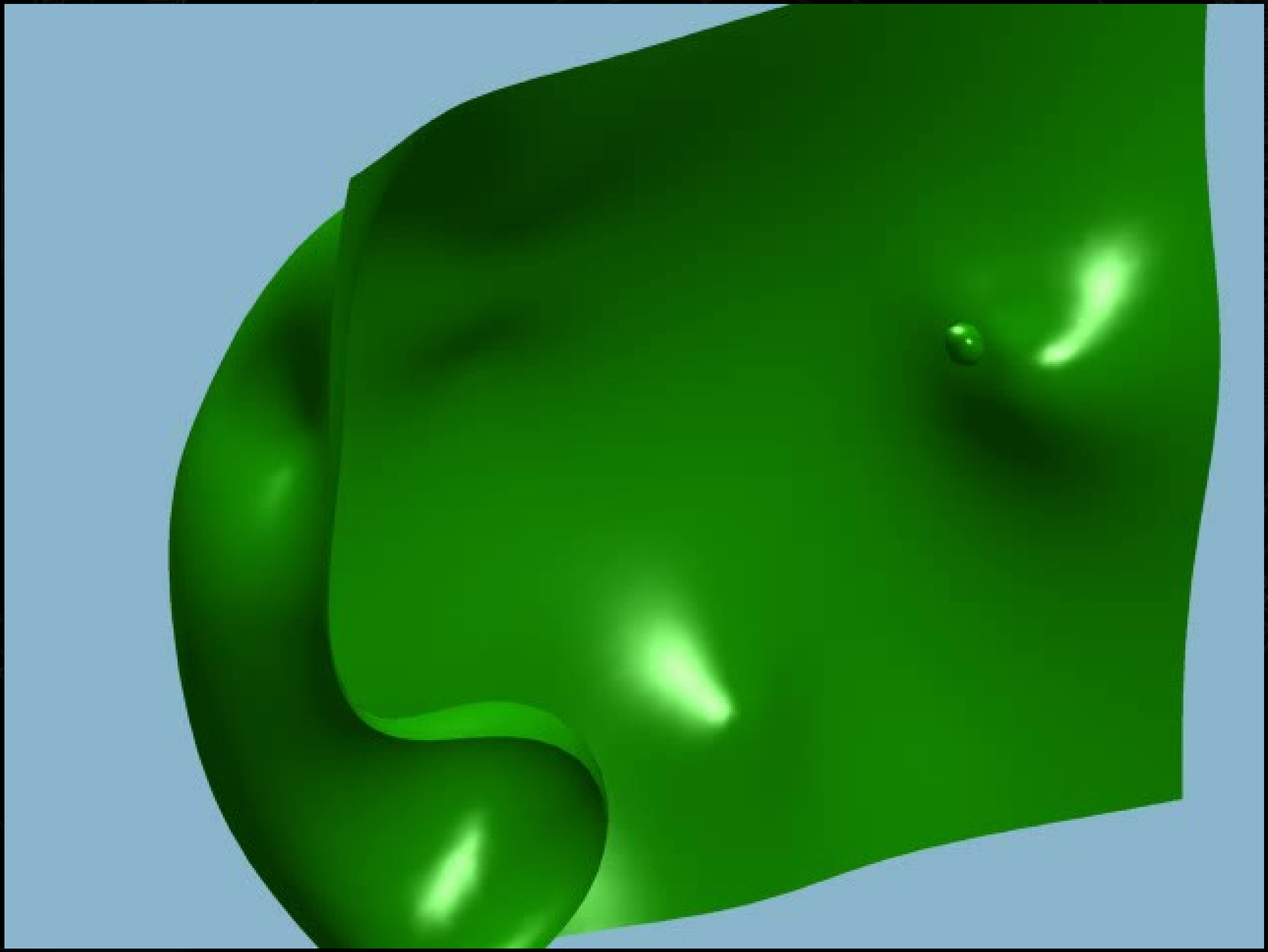
Fitting & evaluation parameters



- Evaluation accuracy << fit accuracy

1D interpolation example

RBF fitting



Results

Dragon

Original mesh	437 645 points 847 414 triangles	15.4MB
New mesh	126 998 points 254 016 triangles	4.5MB
RBF representation	32 461 centres 32 465 coefficients	0.6MB

interpolation points: 872,487 Fit time: 2:51:09

Eval time: 0:04:40

Acknowledgements

- Hand, statue & mannequinn data courtesy of Polhemus corporation
- LIDAR data courtesy of Allen Instruments & Supplies, 1474 Theresa St, Carpinteria, CA93013, USA
- Eros data courtesy NASA & Cornell university
- Buddha & dragon data courtesy of Stanford Computer Graphics laboratory
- All other data courtesy Georgia Institute of Technology